**Angular Interview Preparation**

1. **What is Angular?**

**Sol:** Angular is a TypeScript-based open Source front-end platform, developed and maintained by Google. It offers an easy and powerful way of building front end web-based applications.

**2.What is Angular mainly used for ?**

**Sol:** Angular is typically used for the development of SPA which stands for Single Page Applications.

1. **What is SPA (Single Page Application ) ?**

**Sol:** In the application , if we are trying to navigate to diff Links , It looks we are moving diff pages , But the app has 1 page. The single page dynamically load diff contents in our APP, This is done by JS by Changing the DOM of Single Page at Runtime.

1. **Diff between Angular JS and Angular ?**

|  |  |
| --- | --- |
| It is based on MVC architecture | Uses Components and Directives |
| It uses JavaScript to build the application | Introduced the TypeScript to write the application |
| Based on controllers concept | This is a component based UI approach |
| Not a mobile friendly framework | Developed considering mobile platform |
|  |  |

1. **What are Angular’s Main Concepts ?**

**Sol:**

**Component :** The basic building block of an Angular Application and is used to control HTML Views.

**Modules :**  An Angular module contains basic building blocks like components, services, directives, etc. Using modules you can split your application into logical pieces where each piece performs a single task and is called a “module”.

**Templates :** A Template represents the view of an Angular APP.

**Services:** Service is a class , where we write the Logic to retrieve the data and then we re-use this class in multiple places in App.

**MetaData:** Metadata is used to decorate a class so that it can configure the expected behaviour of a class.

1. **What are Directives ?**

**Sol:** Directives add behaviour to an existing DOM element or an existing component instance.x

import { Directive, ElementRef, Input } from '@angular/core';

@Directive({ selector: '[myHighlight]' })

export class HighlightDirective {

constructor(el: ElementRef) {

el.nativeElement.style.backgroundColor = 'yellow';

}

}

Now this directive extends HTML element behavior with a yellow background as below

<p myHighlight>Highlight me!</p>

1. **What are Components?**

**Sol:** 1**.** Components are the basic building block of an Angular App . A component encapsulates the data , HTML Markup and the Logic for a view behind the view .

1. A component is a class . when we decorate with @component decorator and provide the properties like selector , TemplateUrl,StylesUrl , which makes the class as a Component
2. **Differences Between Component and Directive?**

**Sol**

|  |  |
| --- | --- |
| To register a component we use @Component meta-data annotation | To register directives we use @Directive meta-data annotation |
| Components are typically used to create UI widgets | Directive is used to add behavior to an existing DOM element |
| Component is used to break up the application into smaller components | Directive is use to design re-usable components |
| Only one component can be present per DOM element | Many directives can be used per DOM element |
| @View decorator or templateurl/template are mandatory | Directive doesn't use View |
|  |  |

1. **What are Templates?**

**Sol**: A template is a HTML view where you can display data by binding controls to properties of an Angular component.

**10.What is Data Binding ?**

Data binding is a core concept in Angular and allows to define communication between a component and the HTML DOM Elements ( Templates ) . we can easily interact with application without worrying about how to insert the data. We can make connections in 2 Different ways : 1 way Binding and 2 way Binding.

There are 4 forms of data Binding divided into 3 Categories

1. From the Component to the DOM

**Interpolation: {{ Value }}**

Adds the value of a property from the component.

<li>Name : {{ user.name }} </li>

**Property Binding: [property]=”value”.**

The value is passed from the component to the specified property or an HTML Attribute.

1. From the DOM to the Component

**Event Binding: (event)=”function()”**

When a specified DOM Event happens (click,change,keyup) ,calls the specified method in the component .

<button (click)="logout()"></button>

1. **Two Way Binding : [(ngModel)]=”value”**

Two way Binding allows to have the data flow both ways.

<input type="email" [(ngModel)]="user.email">

**11 . What are Pipes?**

**Sol:** A pipe takes in data as input and transforms it to desired output.

Parameterized Pipe: A pipe can accept any number of optional parameters to fine-tune its output . The Parameterized pipe can be created by declaring the pipe name with colon (:) and then parameter value

{{birthday | date: ‘dd/MM/yyyy’}}

Chaining Pipes: {{birthday | date:’fullDate’ | uppercase}}

Custom Pipe: A pipe is class decorated with @Pipe Decorator

1. The pipe class implements the PipeTransform interface's transform method that accepts an input value followed by optional parameters and returns the transformed value. The structure of pipeTransform would be as below,

interface PipeTransform {

transform(value: any, ...args: any[]): any

return value.substr(0,10);

}

Shorten pipe custom pipe

Pure and impure pipes: